

Test Scenario A:

1. Tests were conducted doing basic gameplay including fighting all enemy types in Region A and Region B.
2. The scenarios completed were as follows on Xbox One (base) test build :
 - a. Launch build numbers in those regions.
 - b. All FX forced to Additive shading model.
 - c. R.detailmode being changed back to 0 (low).
 - d. FX completely disabled

Base Xbox Numbers

Region Name	AVERAGE of FPS	AVERAGE of Frame (ms)	AVERAGE of GT (ms)	AVERAGE of RT (ms)	AVERAGE of GPU (ms)	AVERAGE of DynRes	Time Spent Over Budget (%)	Avg over Budget (ms)
Region A	28.08	35.62	33.02	24.99	29.12	87.97	68.83	38.68
Region A - Additive FX	34.05	29.37	26.08	18.86	26.82	90.31	17.61	37.13
Region A - DetailMode Low	33.84	29.55	26.47	18.25	26.32	96.45	27.39	38.20
Region A - No FX	35.93	27.83	24.46	16.87	26.14	99.94	8.16	38.03
Region B (Region B)	28.89	34.61	29.54	23.44	28.21	90.55	64.99	39.28
Region B - Additive FX	28.72	34.82	30.20	24.25	27.89	89.28	68.29	37.90
Region B - DetailMode Low	31.23	32.02	27.00	23.49	27.73	93.40	37.53	36.69
Region B - No FX	33.03	30.28	25.71	19.38	26.80	99.86	21.71	38.32

Assessment

The no FX scenario illustrates heavy combat and gameplay when the only hindrance is the existing scene and gameplay code, you can subtract that from the base scenario to kind of get an idea of FX's impact to "time over budget". While there's a ton of content-side improvements on the table, there's also systemRegion A issues like gameplay tags can virtually create FX at any time in a completely unrestrRegion Ated manner.

The interesting thing is the biggest drop occurs from Base scenario -> Additive scenario and that provides the biggest gain with the least visual sacrificRegion Ae (compared to going back to DM_Low or no FX). This indRegion Aates that lit translucent FX are a signifRegion Aant performance issue.

Test Scenario B:

3. Tests were conducted at/during the Generic Objective while fighting Generic Boss Guy with a range of different FX settings.
4. The scenarios completed were as follows on all shipped consoles on a test build :
 - a. Base scenario with all default settings.
 - b. All FX forced to Additive shading model.
 - c. R.detailmode being changed back to 0 (low).
 - d. FX completely disabled

Data

PS4								
Scenario Name	AVERAGE of FPS	AVERAGE of Frame (ms)	AVERAGE of GT (ms)	AVERAGE of RT (ms)	AVERAGE of GPU (ms)	AVERAGE of DynRes	Time Spent Over Budget (%)	Avg over Budget (ms)
Base Scenario	33.17	30.15	29.25	15.92	19.98	99.76	38.08	36.93
Additive	34.21	29.23	28.22	15.75	19.95	99.46	21.50	36.30
Detailmode Low	33.23	30.09	29.17	15.70	19.34	99.79	33.07	36.11
No FX	33.68	29.69	29.06	14.69	16.78	100.00	23.30	36.08

PS4Pro								
Scenario Name	AVERAGE of FPS	AVERAGE of Frame (ms)	AVERAGE of GT (ms)	AVERAGE of RT (ms)	AVERAGE of GPU (ms)	AVERAGE of DynRes	Time Spent Over Budget (%)	Avg over Budget (ms)
Base Scenario	36.82	27.16	23.94	13.89	25.89	73.89	4.32	42.22
Additive	35.58	28.10	25.13	13.77	25.59	76.79	5.56	42.88
Detailmode Low	36.70	27.25	23.82	12.66	26.08	80.64	4.91	40.61
No FX	36.53	27.37	24.36	12.50	26.33	86.97	3.37	47.76

Xbox (Base)								
Scenario Name	AVERAGE of FPS	AVERAGE of Frame (ms)	AVERAGE of GT (ms)	AVERAGE of RT (ms)	AVERAGE of GPU (ms)	AVERAGE of DynRes	Time Spent Over Budget (%)	Avg over Budget (ms)

Base Scenario	31.42	31.83	29.78	18.70	27.34	93.50	48.60	36.67
Additive	31.85	31.40	29.27	18.79	27.52	90.09	39.63	36.57
Detailmode Low	31.37	31.88	30.37	17.06	26.46	94.62	51.79	36.94
No FX	33.22	30.11	28.14	16.84	25.66	99.85	22.66	36.26

Xbox One X								
Scenario Name	AVERAGE of FPS	AVERAGE of Frame (ms)	AVERAGE of GT (ms)	AVERAGE of RT (ms)	AVERAGE of GPU (ms)	AVERAGE of DynRes	Time Spent Over Budget (%)	Avg over Budget (ms)
Base Scenario	38.19	26.18	23.38	13.76	24.32	81.20	10.38	39.15
Additive	38.82	25.76	22.64	14.15	24.86	89.10	3.22	39.88
Detailmode Low	38.39	26.05	23.94	12.72	24.28	90.10	8.46	37.17
No FX	39.01	25.63	24.08	12.86	23.74	98.03	6.79	38.30

Assessment

Comparing the results of Test A and B, the biggest hindrance appears to be background env FX eating up too much game thread and GPU - making the impact of the different scenarios far larger than a more “neutral” region despite incredibly heavy gameplay. Meaning, fighting anything remotely heavy in Region A or Region B is far worse than very heavy bounty waves + terrorize in a region with less background costs.

As far as Test B by itself, the Pro and X had DynamRegion A Resolution most affected by the different scenarios. The less FX cost, the closer to 100% dyn res, even though GPU was higher/similar to base scenarios.

In any scenario, switching all FX to “additive” had a tremendous impact on the **Time Spent over budget (%)** in Region A that it would be great to target FX using lit translucency. From observing times that dynamRegion A res would hit minimum values (whRegion A was often on both Base and Pro consoles) it was almost always splashes and boat wakes, especially when swimming “through” them, having them occupy a huge amount of screen space.

Bottom Line

We should hit targeted/highlighted FX that have the biggest impact on performance, and the cost factors that have the biggest impact:

- Remove/reduce env FX in Region B and Region A that are eating up GT and GPU time. If we want to do this we can dig into specific more to make sure we’re not needlessly removing. Those two regions make up 38% of the game’s total environment FX count, and that seems directly correlated with them being the two most expensive regions.
 - The world has ~6200 placed emitters (775 avg if each region was even).
 - Prologue - 533
 - Dead Horse Lake - 1446
 - Region B - 962
 - Deep Ocean - 235
- Change all splashes and boat wakes from something besides Translucent - Default Lit.
 - Of all particle scenarios the biggest drop off was from Base -> Additive vs. DM_Low -> Off. That appeared to have the most significant impact on time over budget.
- Create medium emitters for PS4/Xbox specific Region A ally for splashes/wakes.
- Reduce screen space of cards when way up in the camera (Santiago had some good ideas for this).

Additional Documentation

- Redacted